UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/536,745	05/27/2005	Gregory R Bentz	13210-209	4513

1059 7590 09/30/2009

BERESKIN AND PARR LLP/S.E.N.C.R.L., s.r.l.
40 KING STREET WEST
BOX 401
TORONTO, ON M5H 3Y2
CANADA

EXAMINER	
JORDAN, KIMBERLY L	
ART UNIT	PAPER NUMBER
2194	

MAIL DATE	DELIVERY MODE
09/30/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/536,745	BENTZ ET AL.	
	Examiner	Art Unit	
	Kimberly Jordan	2194	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *28 August 2009*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,2,4,5,7,8,10,11,13,14,16,17 and 19-24* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,2,4,5,7,8,10,11,13,14,16,17 and 19-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This Office action is in response to the amendment filed on August 28, 2009.

2. Claims 1-2, 4-5, 7-8, 10-11, 13-14, 16-17, and 19-24 are pending and have been examined.

3. Claims 22-24 have been added.

4. Claims 1, 5, 7, 11, 13-14, 16-17, and 19-21 have been amended.

5. Claims 3, 6, 9, 12, 15, and 18 have been cancelled.

6. The objection to the specification has been withdrawn in view of Applicant's amendments.

Continued Examination Under 37 CFR 1.114

7. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 08/28/2009 has been entered.

Response to Amendment

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

9. The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

10. Claims 1, 2, 4, 5, 7, 8, 10, 11, 13, 14, 16, 17 and 19-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Swetland (US 2002/0170047 A1)**, in view of **Baentsch et al. (WO 99/49392)**, hereinafter Baentsch, further in view of **Maatta et al. (*Building AS/400 Client/Server Applications with Java*)**, hereinafter Maatta.

As per **Claim 1**, Swetland discloses:

- *a memory unit including executable software* (see at least Figure 7, reference number 750; Paragraph 0046, "The external memory may be used to store programs…");
- *a plurality of class files stored in the memory unit* (see at least Paragraph 0049, "…microprograms and portal data are transmitted from the portal server to the external memory of the portal device.."; "The microprograms in one embodiment are comprised of compact, interpreted instructions known as 'bytecodes,'…"; Paragraph 0077, "As described above, in one embodiment, the bytecodes may be

Java bytecodes/applets."; Paragraph 0083, "…each of the class files used in a particular application program…are combined to form a unified programming object referred to herein as a 'bundle'. For the purpose of illustration, the particular bundle...is constructed from the class files.");

- *a computing unit connected to the memory unit* (see at least Figure 7, reference number 710; Paragraph 0047, "The microcontroller of one embodiment is comprised of a central processing unit ("CPU"), a read only memory ("ROM"), and a scratchpad RAM. The ROM is further comprised of an interpreter module and a toolbox module");
- *the computing unit being able to execute a Java Virtual Machine* (see at least Paragraph 0075, "…the interpreter module on the portal device is a Java virtual machine.");
- *the computing unit configured to execute software [for generating two or more files from the plurality of class files] by combining elements from the plurality of class files without duplication of entries for reducing storage space* (see at least Paragraph 0011, "Accordingly, what is needed is a system and method for reducing the memory requirements for object-oriented programs."; Paragraph 0083, "As illustrated in FIG. 12, in one embodiment, each of the class files used in a particular application program (or applet) are combined to form a unified programming object referred to herein as a "bundle" 1200… More specifically, the redundant MethodRef Foo entries 203,213,223 and 233 are combined into a

single, "global" MethodRef Foo entry 1203 in a shared constant pool 1202 within the bundle 1200.")

- *a constant pool created by combining constant pool entries from two or more of the plurality of class files without duplication of entries* (see at least Paragraph 0083, "…each of the class files used in a particular application program…are combined to form a unified programming object referred to herein as a 'bundle'. For the purpose of illustration, the particular bundle...is constructed from the class files. More specifically, the redundant [method] entries are combined into a single, 'global' [method] entry in a shared constant pool within the bundle." By combining the redundant entries into a unified method, there is no duplication of entries.);
- *a byte codes and information structure created by combining byte codes and information structure entries from the two or more of the plurality of class files* (see at least Figure 2; Paragraph 0084, "The methods and fields from the original class fields are copied to the bundle as well along with various other class file objects (not shown)." Applicant regards "byte codes and information structure" as the "class properties, the methods, fields and attributes of the class, and their types" (Paragraph 0013). The constant pool contains references to these byte codes and information structures. Therefore, if the constant pool entities have been combined into the bundle, the methods, fields, and attributes are copied into the bundle as well.);

- *wherein cross-references between the sibling files in the common sibling group are indicated using hard offsets* (see at least Paragraph 0085, "Accordingly, constant pool entries and other data/code within the bundle reference one another based on an offset value from the top of the constant pool…"), *and*
- *wherein references to files that are not part of the common sibling group are indicated using symbolic references* (see at least Paragraph 0093, "All non-native methods (e.g., those methods containing references to other class files), identified at 1430, are modified in the following manner…At 1440, numeric references identifying local entries are converted into pointers to global entries such as the shared constant pool entries described above. At 1445, certain bytecodes are converted/rewritten into more convenient forms. Finally, at 1450, the method's exception table is converted to references to jop objects instead of numeric references to addresses of bytecodes.")

However, Swetland does not explicitly disclose, but Baentsch discloses:

- *a fixup table for providing information to the Java Virtual Machine for resolving at least one entry in the given generated file at link time* (see at least Figures 1-3; Page 4, lines 8-9, "The cap file also maintains the necessary relocation information in fixup tables."; Page 4, lines 12-13, "The fixup table again contains the position in the text or data section where a relocation has to take place."),

However, Swetland and Baentsch do not explicitly disclose, but Maatta discloses:

- *[the computing unit configured to execute software] for generating two or more files from the plurality of class files [by combining elements from the plurality of*

class files without duplication of entries for reducing storage space] (see at least Page 365, last paragraph, "Split a jar or zip file into smaller jar or zip files.")

- *wherein the number of the generated files is less than the number of the plurality of class files* (see at least Swetland Figure 12; Multiple class files are combined into a bundle file. Maatta shows that the single file may be split into smaller files based on a max file size.),
- *at least two of the generated files are generated as sibling files in a common sibling group, wherein each of the sibling files comprises a sibling list for listing other sibling files in the common sibling group* (see at least Page 370, - split listing),

Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate Baentsch's fixup table into Swetland's combined constant pool and class file technique. The combination allows for additional reductions in memory requirements (Swetland Paragraph 0011). Swetland states in Paragraph 0088 that "Method bytecodes, exception tables, and/or vtable slot numbers associated with the method may also be modified during the bundle generation process to account for the new locations of data within the shared constant pool." Baentsch teaches a fixup table, or relocation table, and Swetland calls for the need of accounting for relocation information because of the bundling process. Baentsch offers an improved method of providing relocation information at link time by using less memory for resource constrained environments. It would have also been obvious to one of ordinary skill in the art at the time the invention was made to incorporate Maatta's jar splitting method into Swetland

and Baentsch's size reduction technique for a bundle of combined java classes. One of ordinary skill in the art is aware that the need for archived files below a max storage size dates back to the use of floppy disks for storage. Zip files were split in order to store the pieces onto multiple floppy disks. This carries through to today with the need for smaller file sizes in handheld devices. WinZip is another software product that provides this functionality.

As per **Claim 2**, the rejection of **Claim 1** is incorporated. However, Swetland and Maatta do not disclose, but Baentsch discloses:

- *wherein the information in the fixup table comprises the location of data needed for resolving a symbolic reference in the given generated file* (see at least Page 4, lines 12-13, "The fixup table again contains the position in the text or data section where a relocation has to take place."; lines 22-24, "…references to other external packages should not be linked by precalculated offsets. Instead, a name or identifier should be used for references to other packages during the link process.")

Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate Baentsch and Maatta into the teachings of Swetland for the reasons listed above.

As per **Claim 4**, the rejection of **Claim 1** is incorporated. However, Swetland and Maatta do not disclose, but Baentsch discloses:

- *wherein for the given generated file, the byte codes and information structure comprises a second hard offset for cross-referencing a method included in the given generated file that was previously symbolically referenced* (see at least Page 4, lines 12-14, "The fixup table...contains the position in the text or data section where a relocation has to take place. In the simple case, these places are also relocated by a precalculated offset into trusted and well known target packages.")

Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate Baentsch and Maatta into the teachings of Swetland for the reasons listed above.

As per **Claim 5**, the rejection of **Claim 1** is incorporated. However, Swetland and Maatta do not disclose, but Baentsch discloses:

- *wherein at least one of the hard offsets does not need to be resolved or put into context by the Java Virtual Machine at link time* (see at least Page 4, lines 12-14, "In the simple case, these places are also relocated by a precalculated offset into trusted and well known target packages."; Page 5, lines 26-27, "The offset of this field can then be precalculated by the converter and need not be linked during the load process.")

Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate Baentsch and Maatta into the teachings of Swetland for the reasons listed above.

As per **Claim 19**, the rejection of **Claim 1** is incorporated. However, Swetland and Maatta do not disclose, but Baentsch discloses:

- *wherein the information in the fixup table of a given sibling files comprises the location of data of a cross-referenced sibling file to place one of the hard offsets that corresponds to the cross-referenced sibling file into context at link time* (see at least Page 4, lines 8-9, "The cap file also maintains the necessary relocation information in fixup tables."; lines 12-14, "The fixup table...contains the position in the text or data section where a relocation has to take place. In the simple case, these places are also relocated by a precalculated offset into trusted and well known target packages."; lines 16-17, "The offset into the target package can be kept…in the fixup table…")

Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate Baentsch and Maatta into the teachings of Swetland for the reasons listed above.

As per **Claim 22**, the rejection of **Claim 1** is incorporated. However, Swetland and Baentsch do not disclose, but Maatta discloses:

- *wherein as few sibling files are generated as possible while satisfying a constraint on individual generated file size* (see at least Page 367, Paragraph , "Splits the source jar or zip file into smaller jar or zip files. No zip entries are added or excluded. The entries in the source jar or zip file are simply distributed among the

destination jar or zip files. The split size is in units of kilobytes (1024 bytes), and specifies the maximum size for the destination files.")

Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate Baentsch and Maatta into the teachings of Swetland for the reasons listed above.

Regarding **Claims 7-8, 10-11, 13-14, 16-17, 20-21, and 23-24**, the scope of the instant claims does not differ substantially from that of Claims 1-5 and 19. Accordingly, Claims 7 and 13 are rejected for the same reasons as set forth in the rejection of Claim 1; Claims 8 and 14 are rejected for the same reasons as set forth in the rejection of Claim 2; Claims 10 and 16 are rejected for the same reasons as set forth in the rejection of Claim 4; Claims 11 and 17 are rejected for the same reasons as set forth in the rejection of Claim 5; Claims 20 and 21 are rejected for the same reasons as set forth in the rejection of Claim 19; and Claims 23 and 24 are rejected for the same reasons as set forth in the rejection of Claim 22.

Response to Arguments

11. Rejection of claims under §103(a):

Applicant's arguments filed August 28, 2009 have been fully considered and are persuasive in part.

Applicant asserts that Baentsch teaches the resolving of references during linking and not for the storage of files. The Examiner agrees that Baentsch's fixup table is used during the linking process as Applicant's claimed fixup table is also. As mentioned in the rejection of

Claim 1, the fixup table is necessary for accounting for the relocation of information during the bundling process of Swetland.

Applicant also asserts that Baentsch's well known and trusted packages as well as external packages are not defined when the file is generated but during the linking process after the file is generated. Swetland is now referenced to teach the referencing of internal as well as external class files, see the rejection of Claim 1.

Applicant asserts that Swetland's bundle is a single file and therefore is not equivalent to the claimed sibling group. The Examiner agrees that Swetland's bundle appears to be drawn to a single file. The above mentioned Maatta reference discloses the splitting of a jar or zip file based on a max file size. Swetland's compressed grouping of class files into a bundle, or single file, is then split according to the teachings of Maatta.

Applicant also asserts that Swetland does not teach that two or more files are generated from the original class files such that there are at least two sibling files. The Examiner agrees, and the Maatta reference was brought in to overcome this deficiency of Swetland.

Applicant asserts that Swetland does not teach providing a sibling list. The Examiner agrees, and the Maatta reference was brought in to overcome this deficiency of Swetland.

Applicant also asserts that there is no need to combine the references of record. The Swetland, Baentsch, and Maatta references are combined for the reasons mentioned in the rejection of Claim 1. Swetland recognizes the fact that relocation information needs to be accounted for because of the bundling process, see at least Paragraph 0085. Baentsch's fixup table accomplishes this in a resource constrained environment.

Applicant finally asserts that there is no need to combine Baentsch's fixup table with Swetland's bundle since Swetland teaches that referencing occurs within the bundle itself and Baentsch teaches referencing external packages. Applicant asserts that these two referencing techniques are mutually exclusive. The Examiner respectfully disagrees. Swetland does mention the fact that references may be made to class files outside the bundle, see Paragraph 0093 and the above rejection of Claim 1. Swetland is not simply directed to referencing class files within the bundle but to external references as well.

Conclusion

12. Any inquiry of a general nature or relating to the status of this application or concerning this communication or earlier communications from the examiner should be directed to Kimberly Jordan whose telephone number is 571-270-5481. The examiner can normally be reached on Monday-Friday 9:30am-5pm EST. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Hyung Sough can be reached on 571-272-6799.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Kimberly Jordan
September 30, 2009
/K.J./
Examiner, Art Unit 2194

/Hyung S. Sough/
Supervisory Patent Examiner, Art Unit 2194
09/28/09